



Universidad Nacional del Nordeste



Facultad de Ciencias Exactas y Naturales y Agrimensura

Licenciatura en Sistemas de Información

Materia: Bases de Datos I

Profesor a Cargo: Lic. Darío Oscar Villegas

Jefe de Trabajos Prácticos: Lic. Walter O. Vallejos

31/10/2023

**Proyecto: Réplicas de Bases de Datos
Comisión 3 – Grupo 9**

Autores:

- Torreani Cáceres, Jimena Soraya
- Villalba, Mauricio Emmanuel
 - Marquez, Marcos Abel
- Romero Ruiz Díaz, Enzo José

Contenido

CAPÍTULO 1: Introducción	3
1.1 Tema	3
1.2 Definición o planteamiento del problema	3
1.3 Objetivos del trabajo práctico	4
1.3.1 Objetivos generales	4
1.3.2 Objetivos específicos	4
CAPITULO 2: Marco referencial	5
CAPÍTULO 3: Metodología seguida	7
3.1 Configurando lo básico para la replicación	7
3.2 Configurando el servidor publicador	8
3.2.1 Activando servicios	8
3.2.2 Configurando distribuidor	10
3.3 Configurando conexión entre servidor publicador y servidor suscriptor	12
CAPÍTULO 4: Desarrollo del tema y presentación de resultados	15
4.1 Creando publicación	15
4.2 Creando un suscriptor	17
4.3 Replicando datos	20
CAPÍTULO 5: Conclusiones	22
CAPITULO 6: Bibliografía	23

CAPÍTULO 1: Introducción

1.1 Tema

El problema en proceso de investigación, desarrollo e implementación en este proyecto, tiene que ver con las técnicas que se utilizan para la replicación de una base de datos. Es decir, poniendo en palabras más simples y concretas, son técnicas que se desarrollan paso a paso para copiar total o parcialmente de forma exacta los datos de una instancia de bases de datos en otras ubicaciones para su posterior utilización, y además para una mejora en la optimización, utilizando un motor de bases de datos.

1.2 Definición o planteamiento del problema

¿A qué nos referimos con REPLICAR una base de datos?

Decimos que una réplica de una base de datos A es otra base de datos B que contiene una copia total o parcial de A, y es actualizada con una periodicidad conocida.

¿Para qué nos puede servir replicar una base de datos?

Replicar bases de datos nos puede servir por ejemplo para:

- Crear Nodos de Contingencia, esto significa que, si se cuenta con una base de datos B en un servidor distinto al de la base de datos A, y esta misma sufre un inconveniente; se podrá seguir operando con los mismos datos en modo lectura o consulta con la base de datos B, ya que, al ser una copia de A, servirá de contingencia (o de respaldo) sin necesidad de interferir la BD A.
- Por cuestiones de proximidad geográfica. Si se necesita consultar una base de datos de una empresa que está distribuida geográficamente, se puede tener nodos de consultas mucho más cercanos de donde se encuentra la sucursal central, sin necesidad de hacer consultas lejanas navegando por distintos tipos de redes.
- Para procesamientos centralizados. Existen empresas que están distribuidas por varias regiones, pero tienen un lugar de almacenamiento común entre todas las sucursales, donde se centralizan todos los datos que se producen en las mismas, obteniendo así una base de datos general con información referida a toda la empresa.
- Escalabilidad y optimización de carga de trabajo. Al tener una réplica de una base de datos tenemos la posibilidad de realizar en ella trabajos pesados como reportes, etc., que no afecten al rendimiento de la base principal.

¿Qué debo tener en cuenta para poder implementar las técnicas de replicación de una base de datos?

En la replicación existen 3 roles; el "Publicador", el "Distribuidor", y el "Suscriptor".

El "Publicador" será el servidor Instancia de SQL server que estará encargado de Publicar la Base de Datos que es la que se va a Replicar a los demás servidores.

El "Distribuidor" será el encargado de distribuir a los suscriptores la Base de Datos que ha publicado el Publicador. Normalmente el Publicador y el Distribuidor se configuran en el mismo Servidor.

El "Suscriptor" será la Instancia que reciba la Réplica del Publicador.

O sea, para llevar a cabo una réplica de base de datos es necesario tener configuradas 2 instancias distintas en el motor de base de datos, una publicadora y otra suscriptora. Y además tener en cuenta que existen dos procesos distintos.

Además de tener configuradas 2 instancias distintas, también debemos tener en cuenta que se trabaja con diferentes versiones de SQL server, porque para que una instancia pueda ser publicadora se necesita que posea cualquier versión de SQL Server excepto la Express.

¿Puedo trabajar con distintos motores de bases de datos?

Sí, se puede trabajar con distintos motores de bases de datos para implementar una replicación de base de datos siempre y cuando estén correctamente configurados.

En este caso particular usaremos sólo el motor de bases de datos SQL SERVER.

¿Cómo es posible implementar esto a partir de una base de datos?

Se trabaja principalmente sobre las configuraciones de las instancias y del motor de base de datos. Más adelante explicaremos con más detalle el proceso de configuración.

1.3 Objetivos del trabajo práctico

Este proyecto de investigación se realizó principalmente con el objetivo de averiguar, entender conceptualmente la replicación de bases de datos, y luego implementarla sobre un modelado de datos.

1.3.1 Objetivos generales

El objetivo general es poder implementar una réplica sobre una instancia de base de datos que actúa como publicadora, la cual ya posee un lote de datos.

1.3.2 Objetivos específicos

El objetivo específico es corroborar que los datos hayan sido replicados en otra instancia de base de datos la cual actuará como suscriptora.

CAPITULO 2: Marco referencial

Luego de una breve definición de lo que es la réplica de bases de datos y de algunas motivaciones o usos que podemos darle a esta tecnología de distribución de datos desde una base de datos a otra, en este capítulo damos un breve ejemplo práctico de aplicación.

En este trabajo de investigación usamos la **replicación transaccional**.

En la replicación transaccional, los cambios de datos que ocurren con frecuencia se automatizan y distribuyen entre servidores.

La replicación de cambios del editor al suscriptor ocurre casi en tiempo real. No se limita a replicar el resultado final de la transacción,

sino que registra cada paso de la transacción y el orden en que ocurren los cambios.

Por ejemplo, en el caso de transacciones en cajeros automáticos, la replicación del editor al suscriptor no es solo el saldo final registrado,

sino todas las transacciones individuales realizadas en el medio. Otra característica clave de la replicación transaccional es que mientras los cambios de datos

en el publicador se replican en el suscriptor, no funciona al revés. Los cambios de datos no se producen en el nivel de suscriptor de forma predeterminada.

Este tipo de réplica permite:

- Tolerar los fallos: si la base de datos principal falla por cualquier motivo, la base de datos que sirve de réplica seguirá estando disponible.
- Mejora la disponibilidad de los datos: gracias a tener una copia actualizada constantemente en una base de datos secundaria que funciona como réplica de la base de datos principal.

Este tipo de réplicas se utiliza principalmente en aquellas operaciones críticas en donde la interrupción del servicio es inaceptable.

Supongamos que tenemos un negocio con un sistema de punto de venta, el cual nos permite llevar un control de ventas e inventarios.

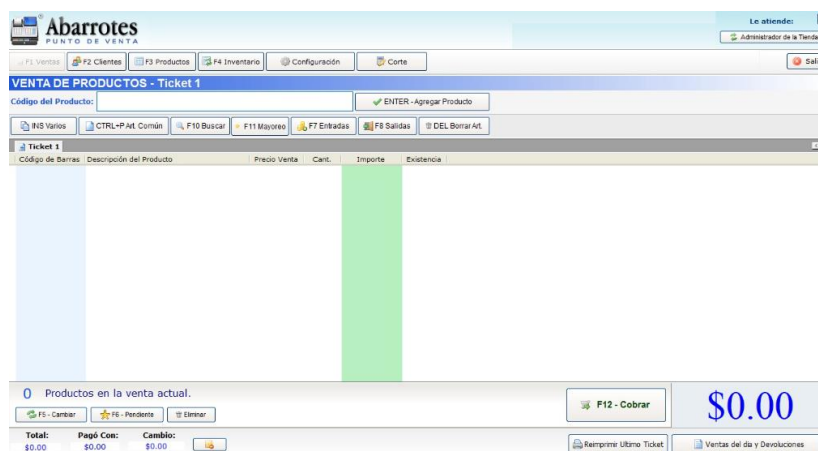


Imagen ilustrativa de un punto de venta

En nuestro negocio vamos a tener nuestra computadora donde cargaremos todas las ventas que se hagan en el día. Cuando un cliente nos compre nosotros cargaremos los datos y, al confirmar la venta, el sistema de punto de venta registrará la transacción en la base de datos principal, que en este caso es la computadora de nuestro negocio.

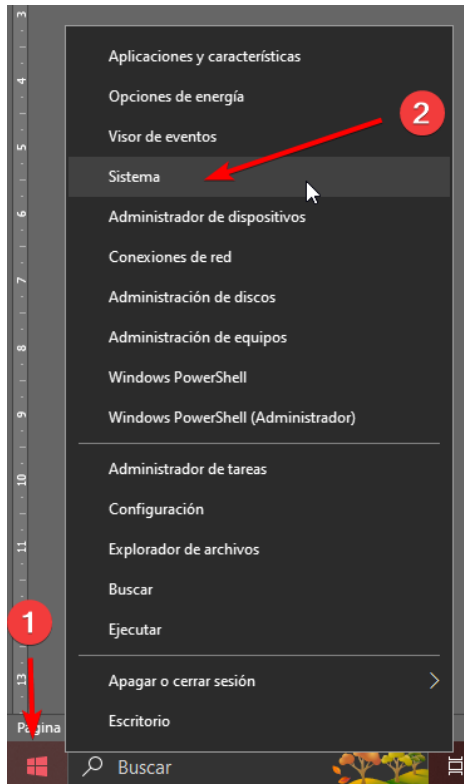
Esta computadora donde nosotros estamos cargando las ventas es crítica, debido a que, si en algún momento esta computadora falla por algún motivo, no podremos seguir vendiendo.

Para solucionar este problema, usaremos una base de datos secundaria que funcionará como copia de seguridad **inmediata** de nuestra base de datos principal del negocio. Esta base de datos secundaria podría estar, por ejemplo, alojada en la nube.

Cuando configuremos nuestra réplica transaccional, la próxima vez que un cliente nos venga a comprar, esta transacción se registrará tanto en la base de datos principal (computadora del negocio) como en la base de datos secundaria (alojada en la nube) de manera casi inmediata con muy poca latencia.

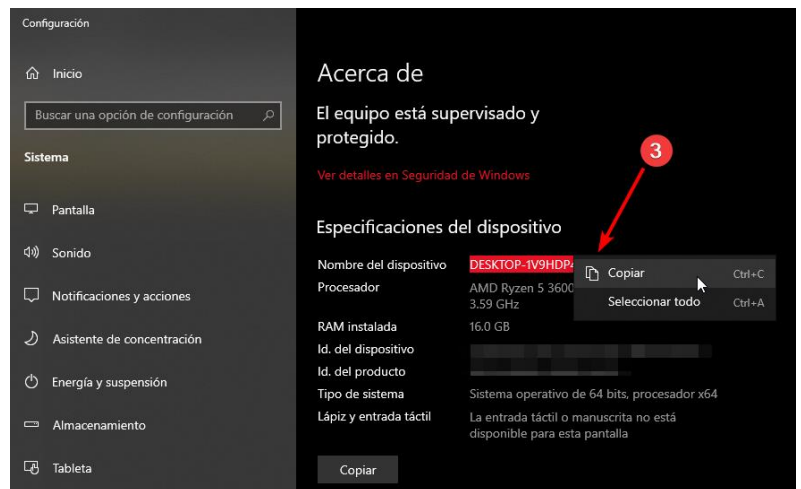
CAPÍTULO 3: Metodología seguida

3.1 Configurando lo básico para la replicación

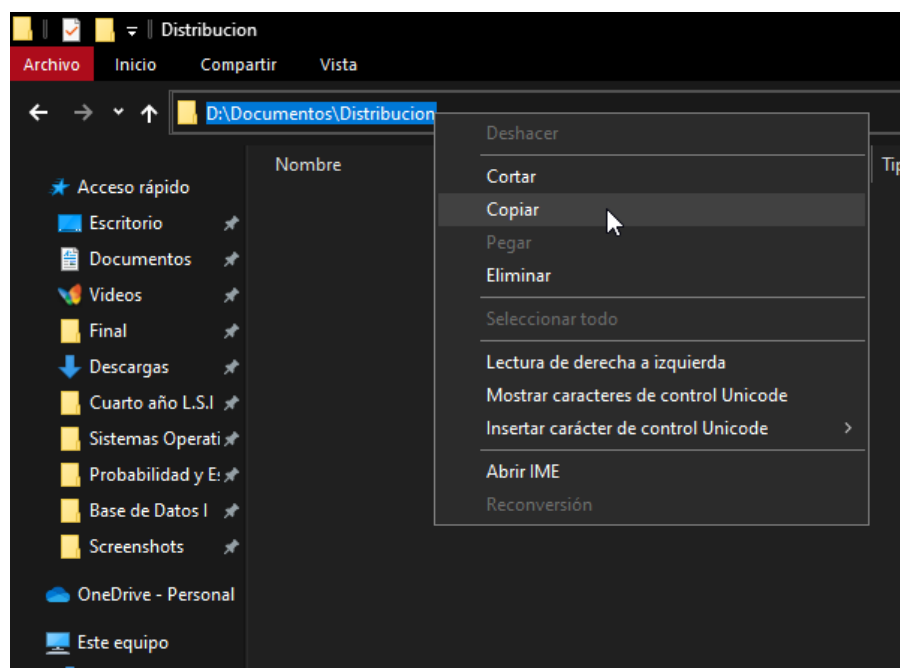


Primero, es muy importante conocer cuál es el nombre de nuestro equipo.

1. Hacemos clic derecho en el botón de Windows
2. Se nos despliegan nuevas opciones. Hacemos clic derecho en **Sistema**
3. Se nos abrirá una nueva ventana con datos de nuestro sistema de Windows. En el apartado **Especificaciones del dispositivo**, aparecerá nuestro **Nombre del dispositivo**. Debemos anotar este nombre para usarlo más adelante.



También debemos crear una carpeta nueva en cualquier parte de nuestro equipo, la cual nos va a servir para la distribución. Una vez creamos esta carpeta, entramos en ella y anotamos la dirección de su ruta para usarla más adelante.



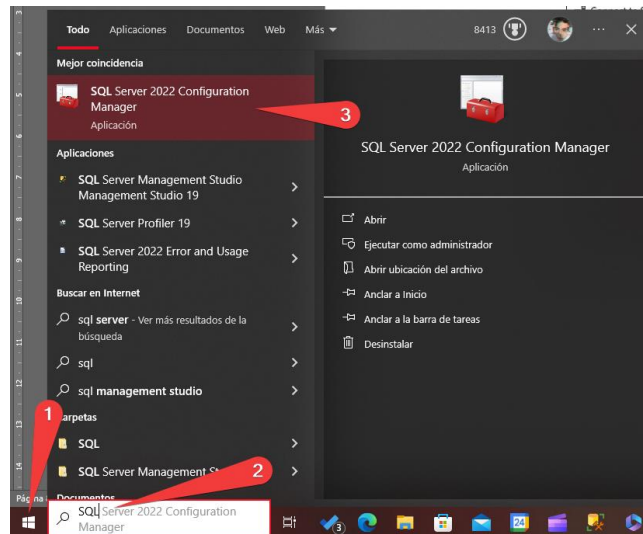
3.2 Configurando el servidor publicador

En esta sección se explicarán los pasos necesarios para **configurar un servidor publicador** para la replicación de una base de datos.

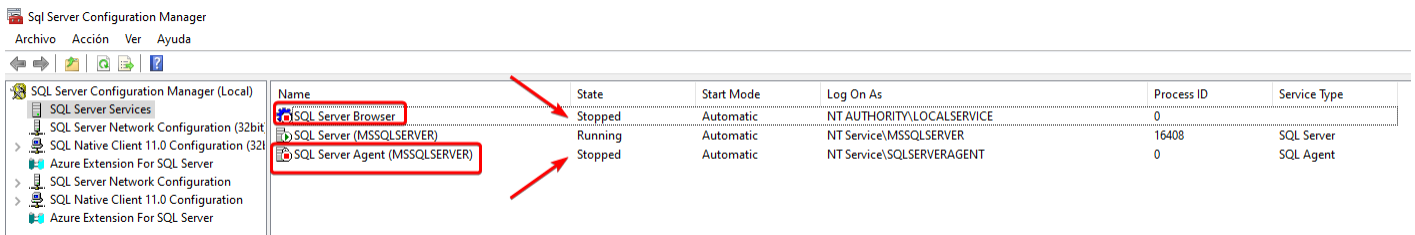
3.2.1 Activando servicios

Es importante comprobar, en primer lugar, si tenemos activados los **servicios de Windows** requeridos para seguir los pasos de esta sección.

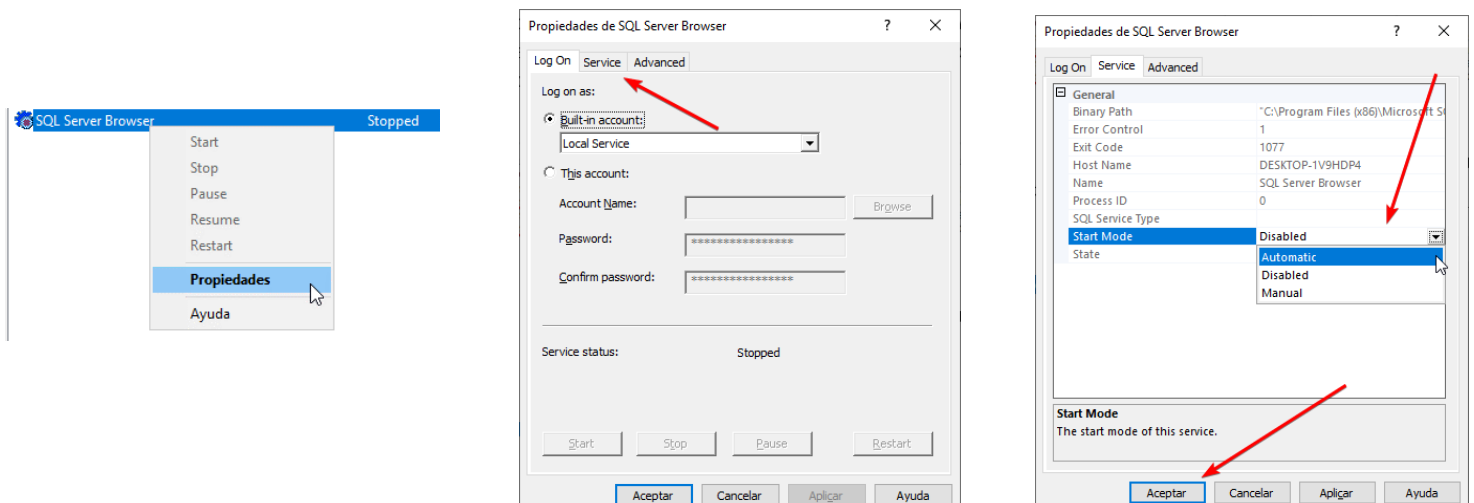
1. Apretamos el botón de **Windows**
2. Buscamos en la barra de búsqueda **SQL**
3. Buscamos y seleccionamos el **Administrador de configuración de SQL Server**



Si es la primera vez configurando un publicador, los servicios **SQL Server Agent** y **SQL Server Browser** no deberían estar funcionando. Debemos activar estos servicios para continuar con la configuración de replicación.

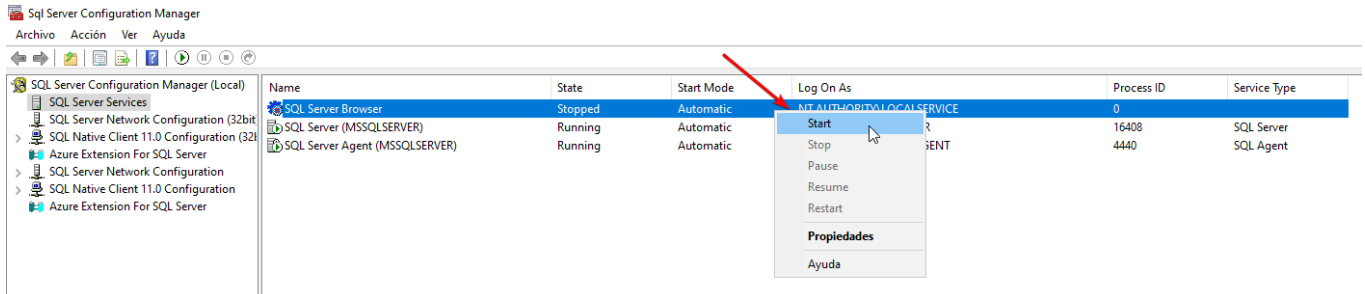


Hacemos clic derecho en uno de los servicios marcados con rectángulo rojo, seleccionamos **Propiedades**. Se nos abrirá una nueva ventana, seleccionamos la pestaña **Service**. Se nos abrirá una nueva ventana, seleccionamos en **Disabled** y marcamos la opción **Automatic**. Por último, apretamos el botón **Aceptar** y cerramos las ventanas del servicio.

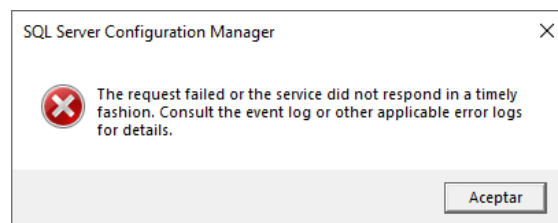


Este mismo procedimiento debe realizarse para ambos servicios: **SQL Server Agent** y **SQL Server Browser**.

Podremos observar en la columna **Start Mode** que ahora ambos servicios están en **Automatic**. Seleccionamos con clic derecho uno de los servicios, y luego en la opción **Start** para iniciarlo. Hacemos lo mismo con ambos servicios.



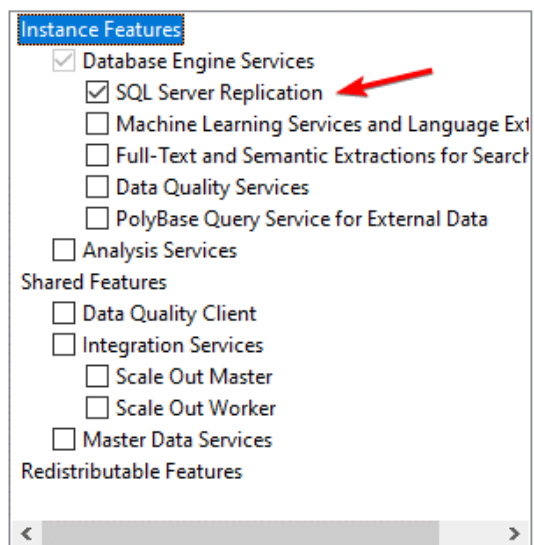
SI POR ALGÚN MOTIVO SALE EL SIGUIENTE MENSAJE DE ERROR AL INTENTAR INICIAR EL SERVICIO DE SQL SERVER AGENT, ASEGURARSE QUE LA VERSIÓN INSTALADA DE SQL SERVER NO SEA LA “EXPRESS” O “COMPACT”. DE LO CONTRARIO, NO SE PODRÁ CONTINUAR, YA QUE ESTAS VERSIONES NO SOPORTAN SQL SERVER AGENT.



En caso de no poder solucionar este error por ningún método, desinstalar la versión SQL Server Express o Compact e instalar cualquier otra versión excepto SQL Server Express y SQL Server Compact.

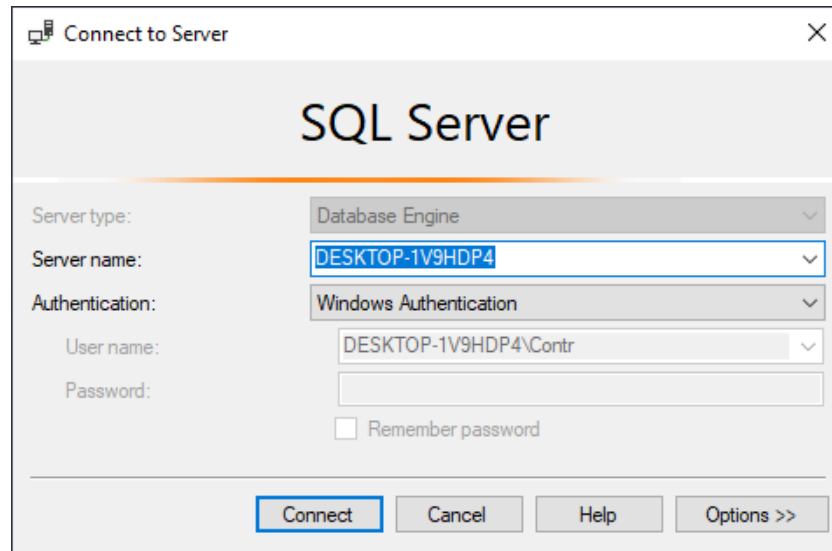
Al ejecutar el instalador del SQL Server, realizar una instalación “Personalizada” para marcar la opción SQL Server Replication. La opción “instalación básica” no trae incluida esta opción.

Features:

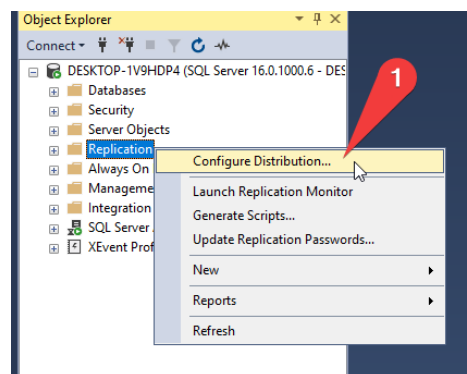


3.2.2 Configurando distribuidor

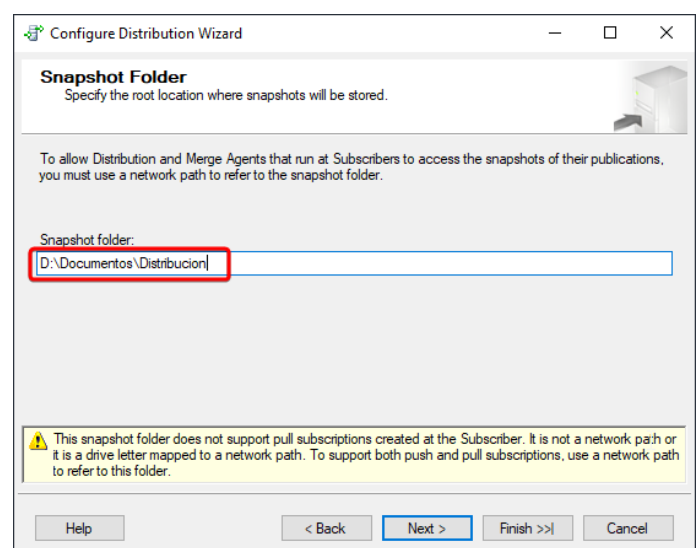
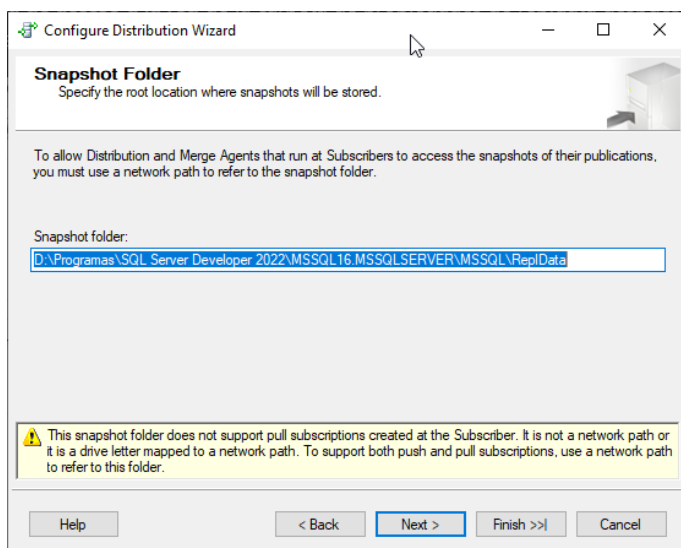
Una vez inicializado los servicios necesarios, abrimos SQL Server. Debemos conectarnos al servidor que será el **publicador**, ya sea mediante **autenticación Windows** o **autenticación de servidor SQL**. En nuestro caso, al ser un servidor local, es el mismo nombre de nuestro equipo que teníamos anotado.



1. Una vez conectado a nuestro servidor, hacer clic derecho en la carpeta **Replication** y seleccionar la opción **Configure Distribution...**



Se abrirá el agente de configuración, apretaremos el botón **Siguiente** hasta llegar a la sección de **Snapshot Folder**. La dirección que viene de forma predeterminada debemos cambiarla por la ruta que anotamos anteriormente de la carpeta **Distribución**.



Colocamos el nombre que queramos y apretamos el botón **Next >** hasta llegar al botón **Finish**, presionamos el botón **Finish**.

Configure Distribution Wizard

Distribution Database
Select the name and location of the distribution database and log files.

The distribution database stores changes to transactional publications until Subscribers can be updated. It also stores historical information for snapshot and merge publications.

Distribution database name:
distribucion_base_consortio

Folder for the distribution database file:
D:\Programas\SQL Server Developer 2022\MSSQL16.MSSQLSERVER\MSSQL\Data

Folder for the distribution database log file:
D:\Programas\SQL Server Developer 2022\MSSQL16.MSSQLSERVER\MSSQL\Data

The paths must refer to disks that are local to the Distributor and begin with a local drive letter and colon (for example, C:). Mapped drive letters and network paths are invalid.

Buttons: Help, < Back, Next >, Finish >>, Cancel

Cuando termine de crearse la distribución, nos saldrá una ventana de **Success**.

Configure Distribution Wizard

Configuring...
Click Stop to interrupt the operation.

Success

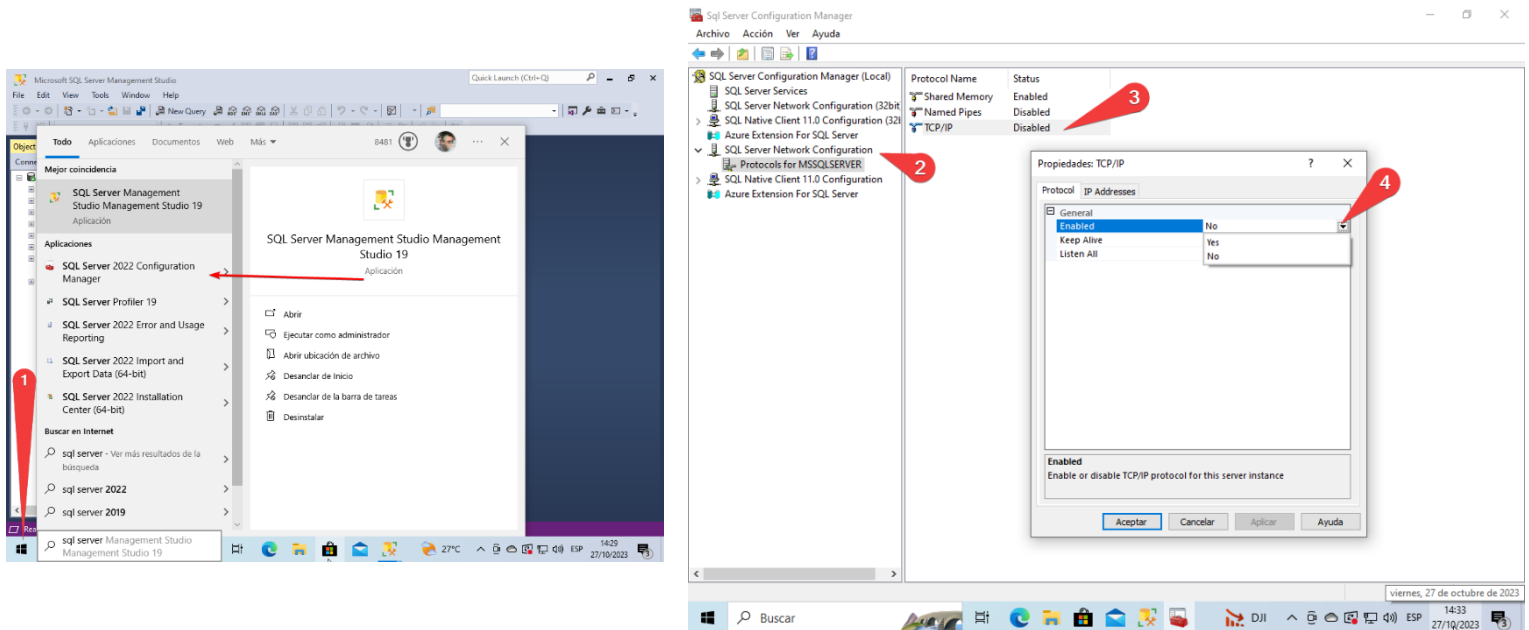
Action	Status	Message
Configuring the Distributor	Success	
Enabling Publisher 'DESKTOP-1V9HDP4'	Success	

Buttons: Stop, Report, Close

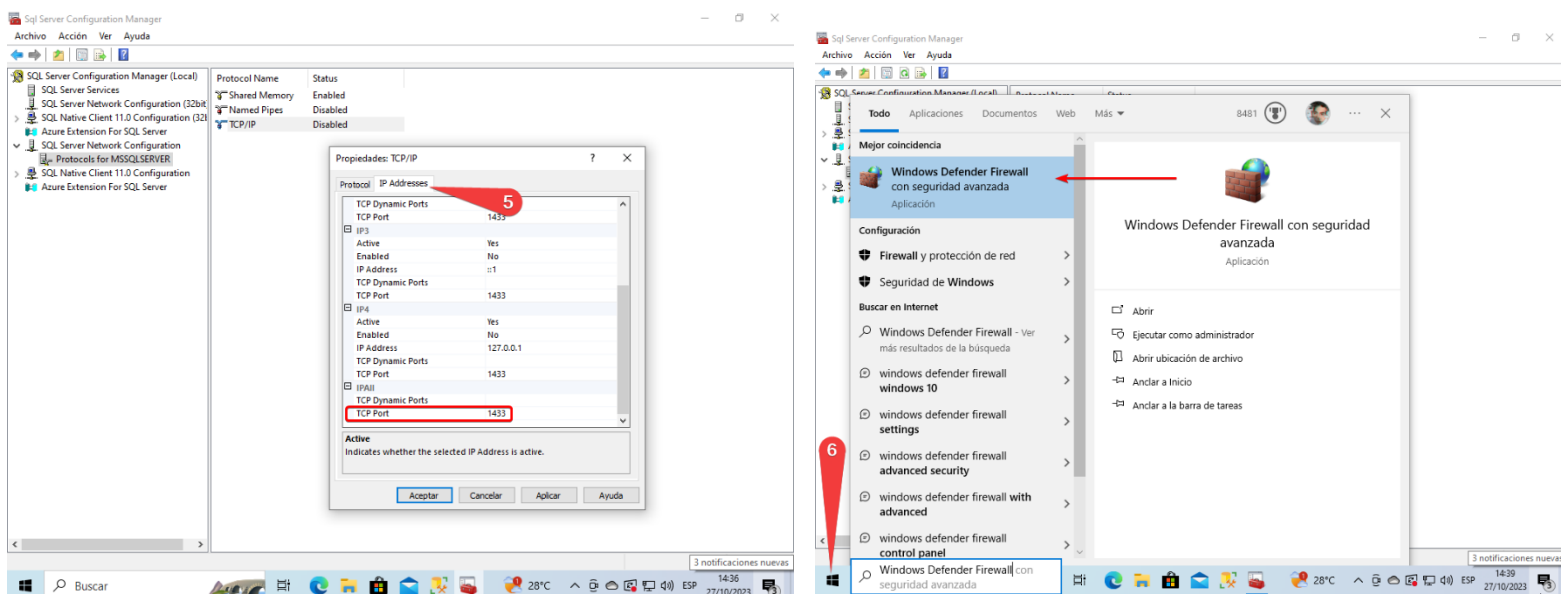
3.3 Configurando conexión entre servidor publicador y servidor suscriptor

Debido a que nuestro proyecto decidimos usar el servidor suscriptor dentro de una máquina virtual, es requisito obligatorio seguir los pasos que se explicarán a continuación.

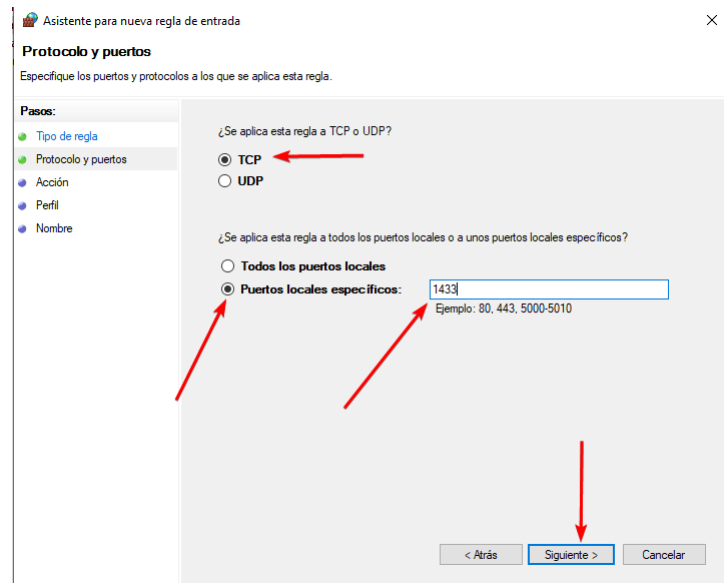
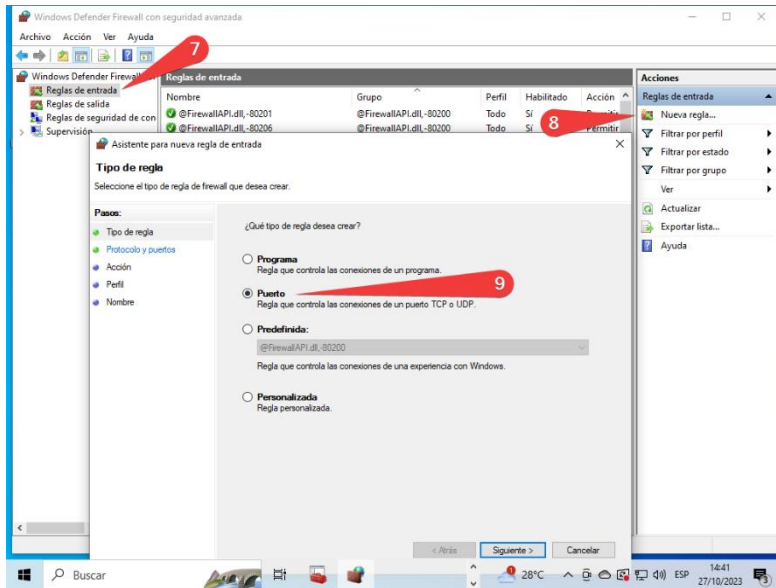
1. En la máquina virtual buscamos y abrimos **SQL Server Configuration Manager**
2. Desplegamos la pestaña **SQL Server Network Configuration** y seleccionamos **Protocols for MSSQLSERVER**
3. Hacemos doble clic en **TCP/IP**
4. Desplegamos la pestaña **Enabled** y seleccionamos **Yes**. Lo mismo para la pestaña **Listen All**, debemos asegurarnos que también esté marcado en **Yes**



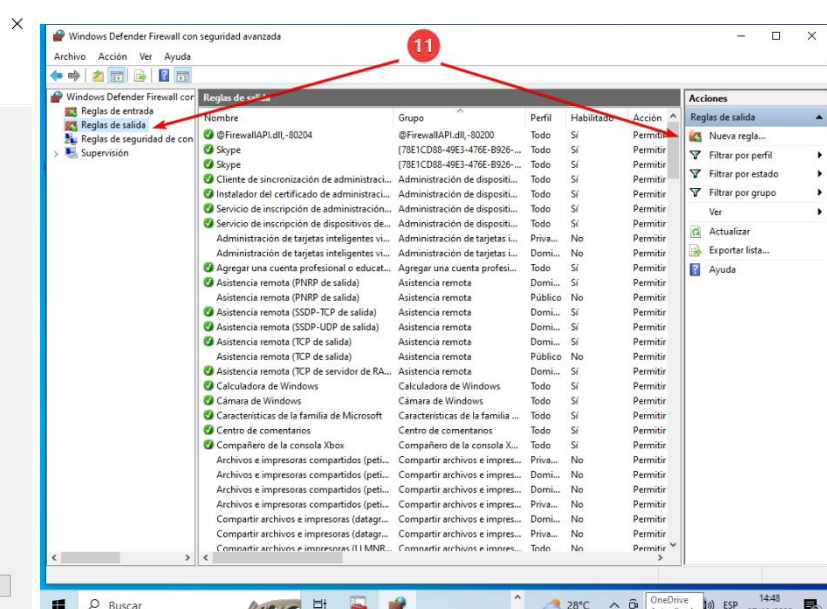
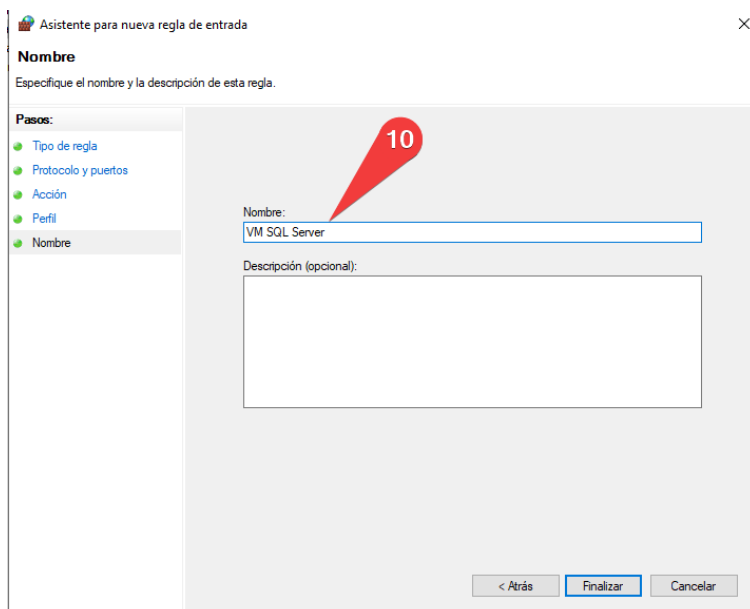
5. Nos dirigimos a la pestaña **IP Addresses**, bajamos hasta el final de la ventana y anotamos el número del **TCP Port**. Por lo general es 1433. Presionamos **Aceptar**
6. Presionamos el botón de **Windows**, buscamos y seleccionamos **Windows Defender Firewall con seguridad avanzada**



7. Seleccionamos en el menú izquierdo **Reglas de entrada**
8. Seleccionamos en el menú derecho **Nueva regla...**
9. Marcamos la opción **Puerto** y presionamos **Siguiente >**. Marcamos las opciones **TCP**, luego **Puertos locales específicos** y en el cuadro de texto ponemos nuestro puerto **TCP** que anotamos anteriormente. Presionamos el botón **Siguiente >**. Seleccionamos la opción **Permitir la conexión** en la sección de **Acción**. En la sección **Perfil** mantendremos las opciones marcadas por defecto: **Dominio, Privado y Público**



10. Presionamos **Siguiente** hasta llegar a la sección donde colocaremos el nombre a nuestra nueva regla. El nombre puede ser cualquiera. Por ejemplo: **VM SQL Server**. Presionamos el botón **Finalizar**
11. Los pasos 8, 9 y 10 deben realizarse de igual forma para **Regla de salida**. El nombre que pusimos para la regla de entrada debe ser la misma que el nombre para la regla de salida, en este caso, **VM SQL Server**



Estos pasos anteriormente explicados fueron realizados en la **máquina virtual** que usaremos para el **suscriptor**. **Los mismos pasos del 1 al 11 deben ser realizados en la máquina “física” o la computadora que será el servidor publicador.**

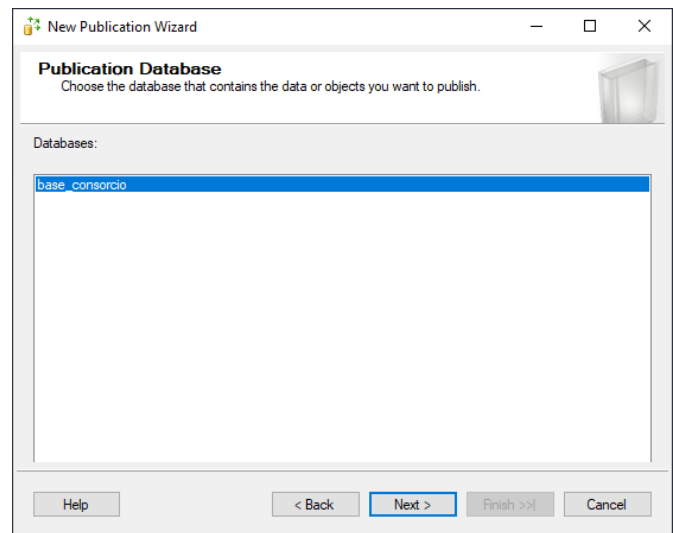
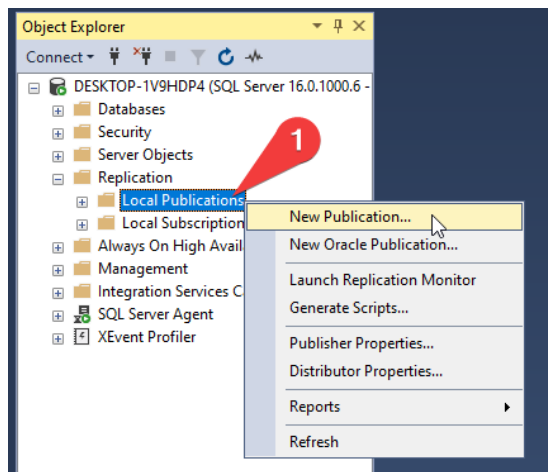
IMPORTANTE: Una vez configurado las reglas del firewall del servidor publicador y del servidor suscriptor, debemos asegurarnos que, tanto la máquina virtual del servidor suscriptor como la máquina física del servidor publicador, cuenten con un **inicio de sesión mixto: mediante SQL Server y autenticación de Windows**. Por lo general, el usuario predeterminado es **sa** cuando instalamos **SQL Server**.

CAPÍTULO 4: Desarrollo del tema y presentación de resultados

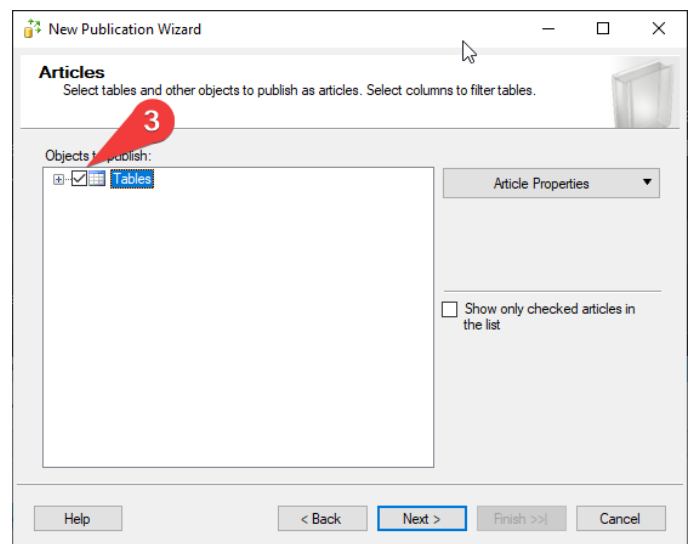
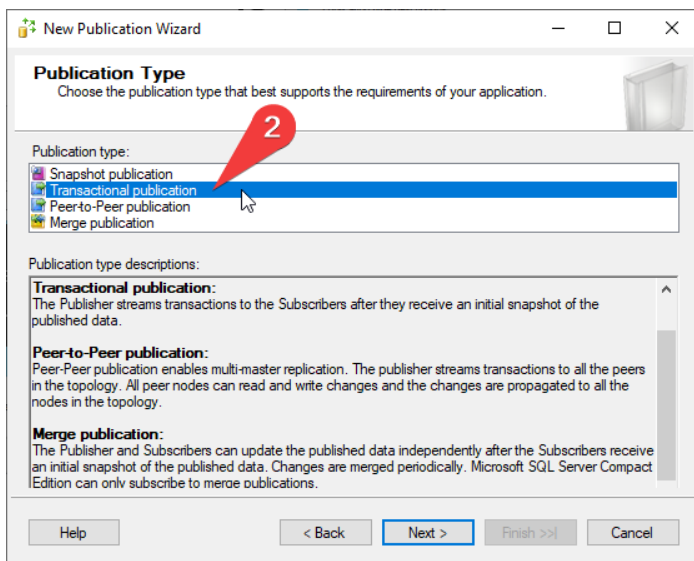
Procederemos a crear una publicación transaccional para replicar una base de datos. En nuestro caso, **base_consortio**. Para realizar estos pasos, debemos estar conectados al **servidor publicador** que configuramos previamente en nuestro manejador SQL.

4.1 Creando publicación

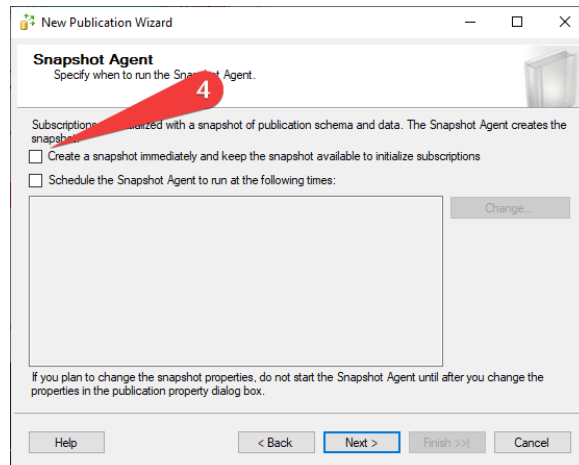
1. Desplegamos la pestaña **Replication**, hacemos clic derecho en **Local Publications** y seleccionamos la opción **New Publication...** Seleccionaremos nuestra base de datos **base_consortio** y presionamos **Next >**



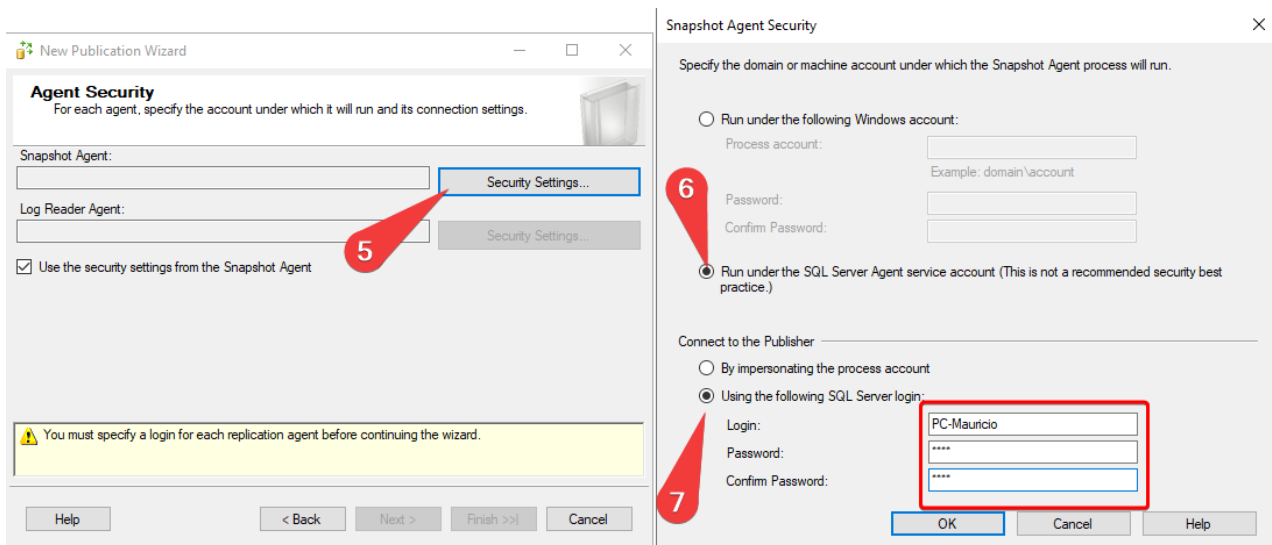
2. Seleccionamos en **tipo de publicación**, la **Transaccional publication**. Presionamos **Next >**
3. Marcamos la opción **Tables**. Esto replicará todas las tablas de **base_consortio**. También podemos desplegar esta pestaña y seleccionar de forma personalizada qué tablas de **base_consortio** vamos a replicar. Presionamos **Next >**



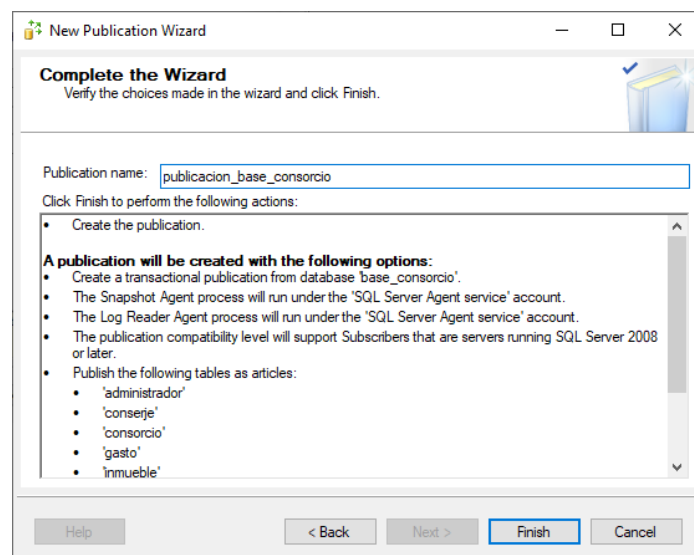
4. Cuando lleguemos a esta ventana, marcamos la opción **Create a snapshot immediately**. Presionamos **Next >**



5. Apretamos el botón **Security Settings...**
6. Marcamos la opción **Run under the SQL Server Agent service Account**
7. Marcamos la opción **Using the following SQL Server Login**, y completamos los datos con las credenciales de una cuenta creada administradora en nuestro **servidor publicador**.



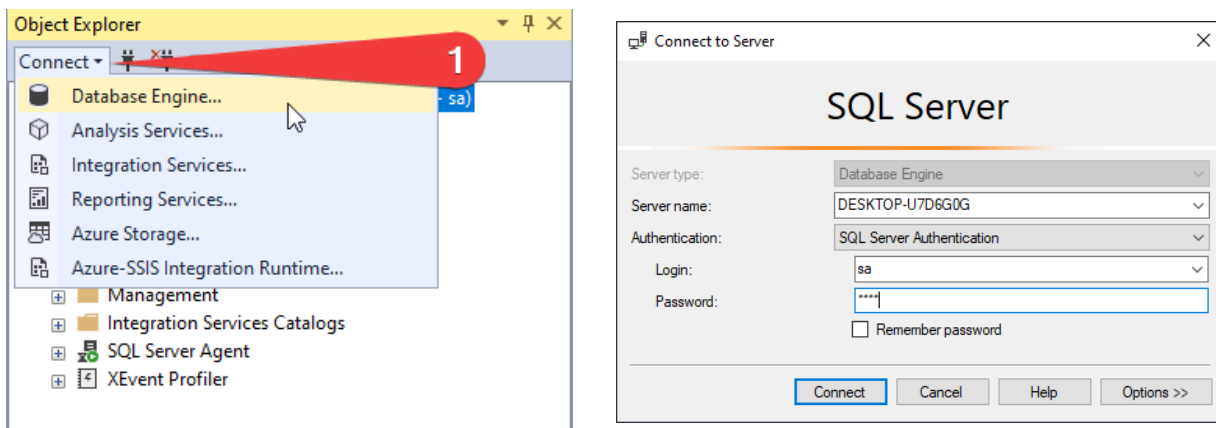
Presionamos el botón **Siguiente** hasta llegar a la ventana donde pondremos el nombre a la publicación. Ponemos un nombre cualquiera. Presionamos luego **Finish**, y ya tendremos nuestra publicación creada.



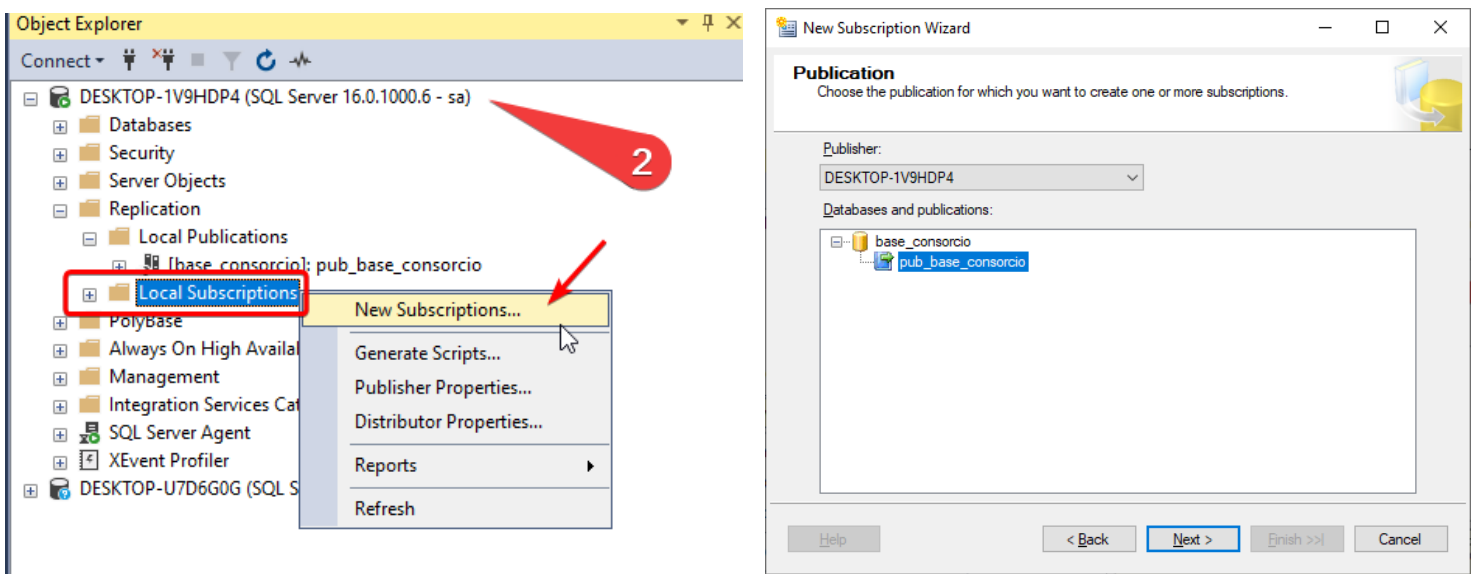
4.2 Creando un suscriptor

En esta sección vamos a crear un suscriptor. Para ello, necesitamos tener instalado SQL Server en otra computadora que esté conectada a nuestra red local o instalar una máquina virtual en nuestra computadora. En este caso, por practicidad, usaremos una **máquina virtual** como [VirtualBox](#). Los pasos para conectar la máquina física con la máquina virtual ya fueron explicados en la sección 3.3.

1. En nuestra máquina física del servidor publicador, presionamos el botón **Connect** en el manejador SQL y seleccionamos la opción **Database Engine...** Se nos abrirá una nueva ventana y colocamos en **Server name** el nombre del equipo suscriptor. Usamos la autenticación **SQL Server** para conectarnos a la base de datos del equipo suscriptor.

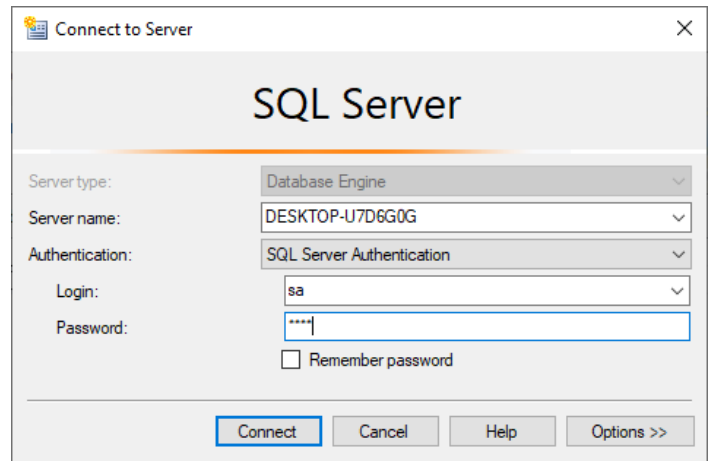
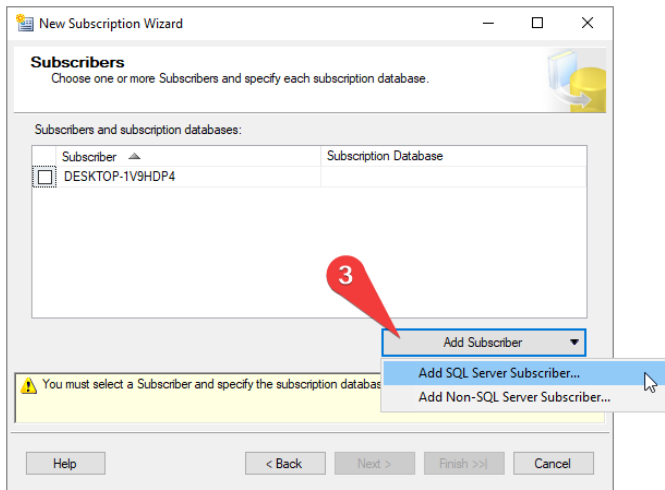


2. En la base de datos del equipo publicador, desplegamos la pestaña **Replication**, hacemos clic derecho en **Local Subscriptions** y seleccionamos **New Subscription**. Si realizamos la publicación correctamente, ya deberá aparecernos en la ventana de **Publication**. Presionamos **Next >**

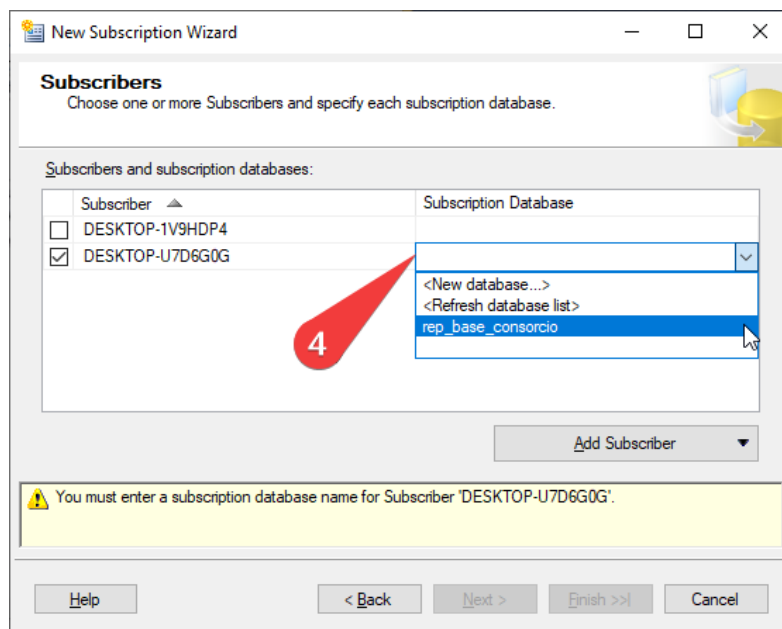


Presionamos siguiente hasta llegar a la ventana donde agregaremos al servidor suscriptor

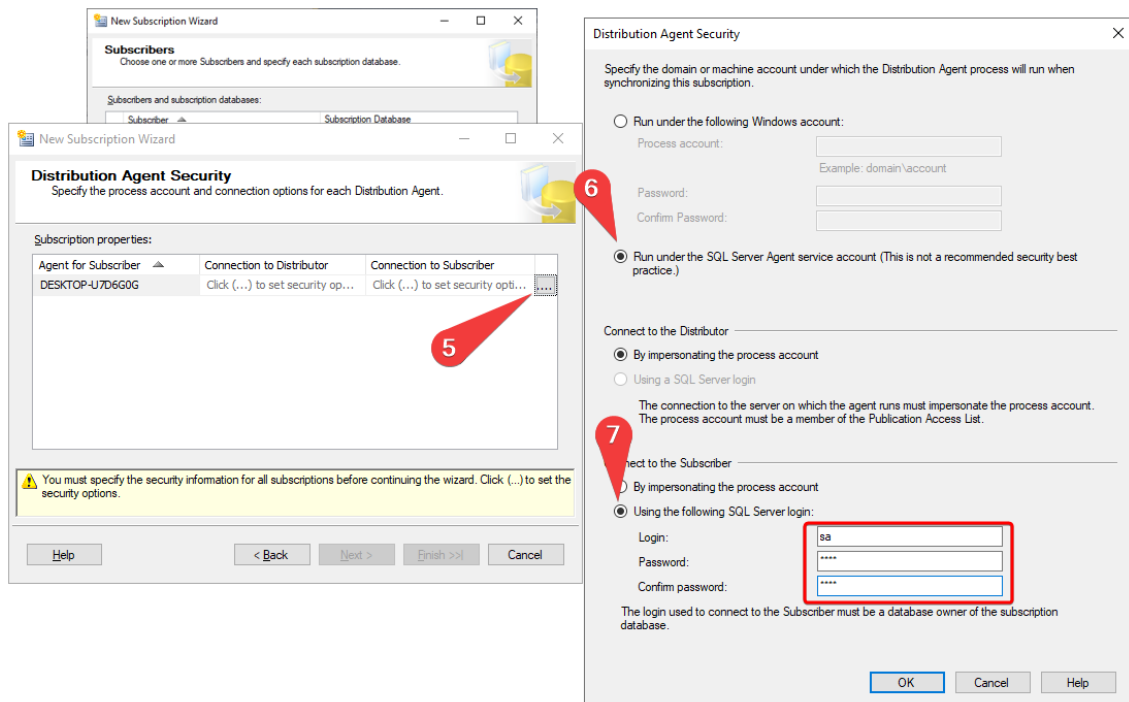
3. Apretamos el botón **Add Subscriber** y seleccionamos la opción **Add SQL Server Subscriber...** En la siguiente ventana, en **Server name** escribiremos el nombre del equipo del servidor suscriptor. Usaremos la **autenticación de SQL Server** y colocaremos los datos de inicio de sesión del servidor suscriptor.



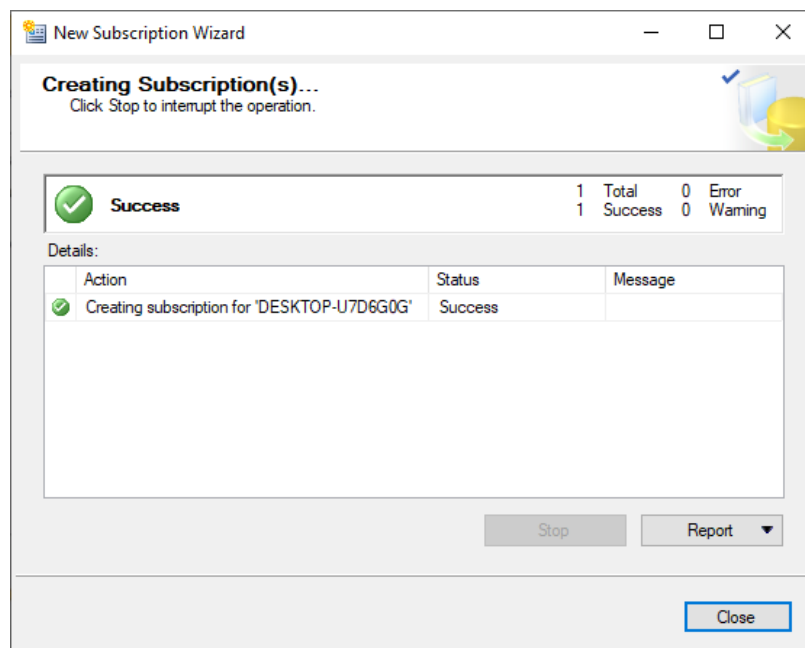
4. Nos aparecerá en la lista nuestro equipo suscriptor. Desplegamos la pestaña de **Subscription Database**, y seleccionamos la base de datos donde los datos van a ser replicados. Presionamos **Next >**



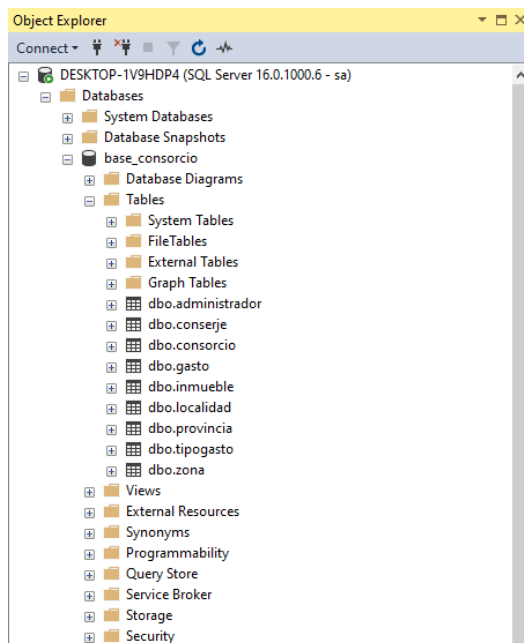
5. Seleccionamos los tres puntos
6. Marcamos la opción **Run under the SQL Sever Agent service Account**
7. Marcamos la opción **Using the following SQL Server login**. Luego, completamos las credenciales de inicio de sesión del servidor suscriptor mediante **autenticación SQL**. Cerramos esta ventana apretando el botón **OK**. Luego presionamos el botón **Next >** hasta finalizar, manteniendo todas las opciones siguientes de forma predeterminada.



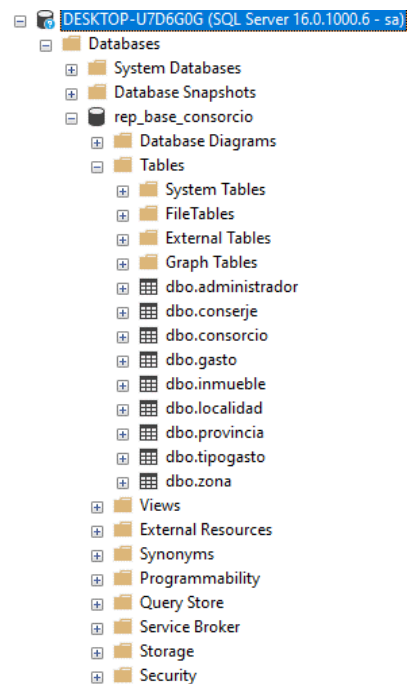
Si está todo configurado correctamente, nos saldrá la siguiente ventana diciéndonos que la suscripción se creó correctamente.



Podremos observar en el explorador de objetos de nuestra máquina física que, si desplegamos la pestaña **Tablas**, tanto de **base_consortio** en el servidor publicador, como de **rep_base_consortio** en el servidor suscriptor, tendremos exactamente las mismas tablas.



Servidor publicador

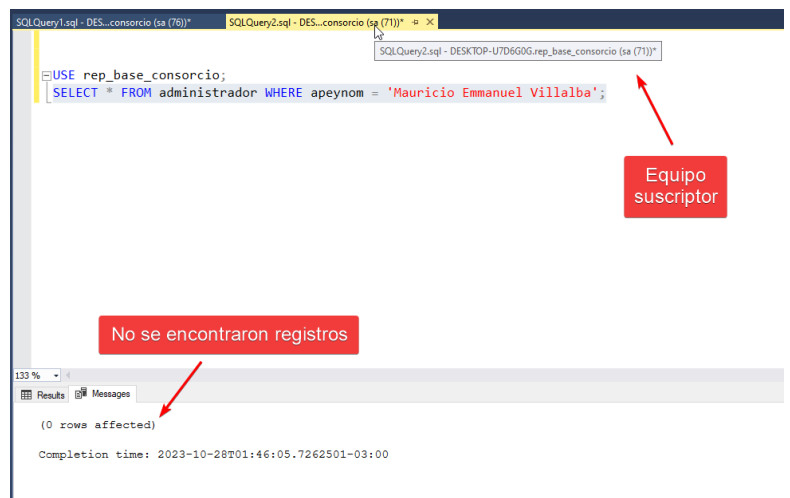
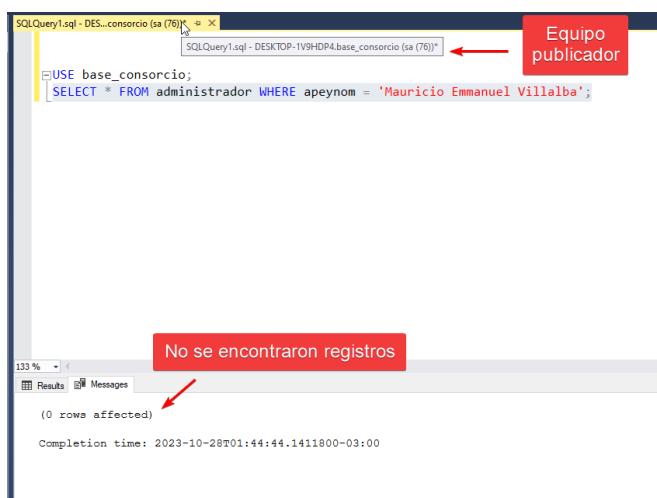


Servidor suscriptor

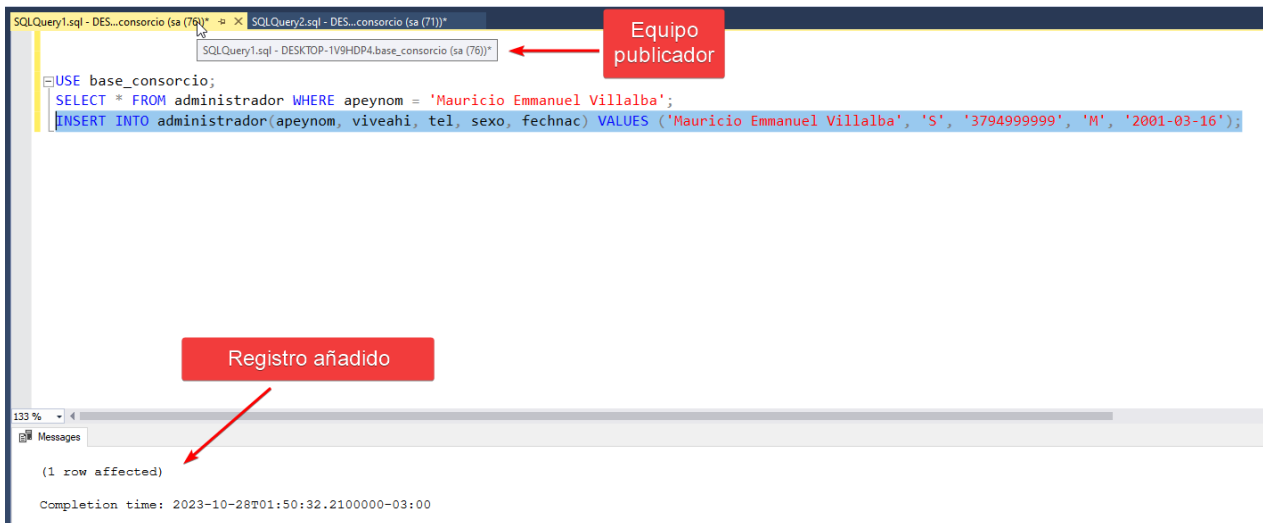
4.3 Replicando datos

Ahora que tenemos la suscripción hecha a la publicación de nuestra base de datos **base_consortio** perteneciente al equipo publicador, podemos comenzar a replicar los datos en tiempo real.

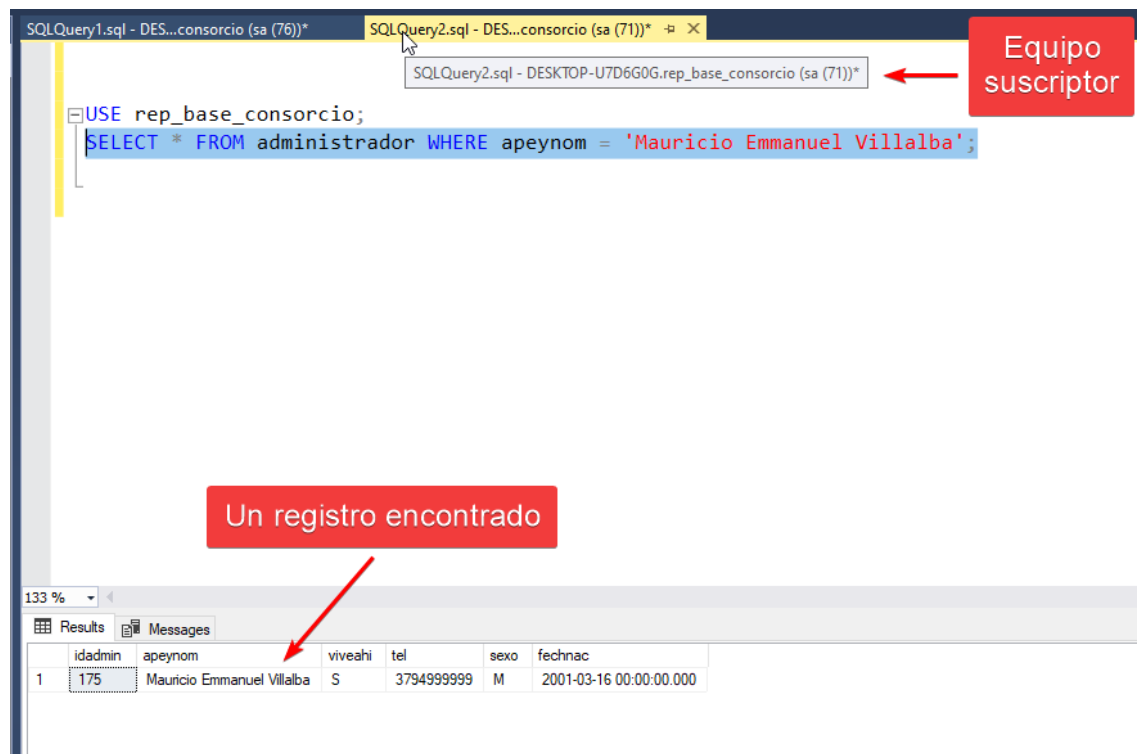
Primero, verificamos que un registro no existe. Por ejemplo, el administrador “Mauricio Emmanuel Villalba” no está como registro en la tabla **administrador**. Lo verificamos tanto en **base_consortio** del equipo publicador como en **rep_base_consortio** del equipo suscriptor.



En **base_consortio** del equipo publicador, cargaremos el administrador 'Mauricio Emmanuel Villalba'



Ahora, en **rep_base_consortio** del equipo suscriptor realizamos la misma consulta para ver si encuentra el registro.



De esta manera queda comprobado que la replicación transaccional funciona correctamente. Cualquier otro tipo de modificación ya sea actualización o eliminación de registros y agregar, modificar o eliminar tablas en la base de datos **base_consortio**, se vería reflejado correctamente en la base de datos replicada **rep_base_consortio**.

CAPÍTULO 5: Conclusiones

Observamos que las réplicas son una tecnología muy potente e importante para las operaciones críticas del día a día. En nuestro caso, al trabajar específicamente con el tipo de réplicas transaccionales, podemos concluir con algunos beneficios que esta nos aporta, las cuales además de mejorar la disponibilidad de los datos y la tolerancia a los fallos, nos ofrece otras ventajas:

- Nos garantiza que nuestro servicio se mantenga en línea, con una fuente de datos alternativa, en caso de que falle nuestra base de datos principal.
- Es transparente, ya que, al tener coherencia de datos en ambos servidores, en caso de que falle uno, automáticamente estaremos usando una fuente de datos alternativa sin darnos cuenta.
- Esto mejora el rendimiento ya que distribuye la carga de trabajo entre servidores, mejorando tiempos de respuesta, ya que las consultas pueden dirigirse a la base de datos réplica sin afectar el rendimiento de la base de datos principal.
- Permite tener réplicas en distintas ubicaciones geográficas.
- La replicación de datos es una excelente manera de proporcionar un acceso constante a los datos.
- También aumenta el acceso a los datos a múltiples usuarios al mismo tiempo.
- Las redundancias de datos se eliminan mediante la fusión de bases de datos y la actualización de bases de datos esclavas con datos incompletos.
- Y con la replicación de datos, naturalmente, hay un acceso más rápido a las bases de datos.

También destacamos algunas desventajas que observamos a la hora de aplicar réplicas transaccionales:

- En nuestro proyecto realizamos el caso donde el suscriptor funciona en modo “solo lectura”. Esto quiere decir que el suscriptor recibe los datos casi inmediatamente actualizados por el publicador, sin embargo, no puede escribir sobre la base de datos.
- Para que tanto el publicador como el suscriptor puedan realizar actualizaciones en la base de datos replicada, agrega mucha complejidad a la sincronización y a la coherencia de datos entre publicador y suscriptor.
- Esto implica que, configurar y mantener una réplica transaccional puede llegar a ser muy complicado y se requiere diseñarlo cuidadosamente y monitorearlo constantemente para garantizar la consistencia de datos y la sincronización adecuada.
- Si bien nuestro proyecto fue realizado con una máquina virtual para simular el suscriptor, es probable que este tipo de tecnologías se usen en distintos servidores físicos. Esto conlleva a costos adicionales para comprar hardware adicional.
- La replicación de datos requiere un gran espacio de almacenamiento e infraestructura para mantenerla.
- También abre más partes del software a violaciones de privacidad y seguridad.

CAPITULO 6: Bibliografía

Sitios de consulta utilizados:

[Replicación de SQL Server - SQL Server | Microsoft Learn](#)

<https://www.tibco.com/es/reference-center/what-is-data-replication>

<https://www.astera.com/es/type/blog/data-replication/>

<https://www.ibm.com/docs/es/rational-clearquest/9.0.0?topic=multisite-database-replication-creation>